Attorney Docket: 30566.112USU1

Patent Application

for

METHOD AND APPARATUS FOR OBTAINING A SET OF MAPS

Inventor(s):

HOWARD MARANTZ
NEMMARA CHITHAMBARAM
SCOTT DING

Prepared By:

Jason S. Feldmar
Gates & Cooper
Howard Hughes Center
Suite 1050
6701 Center Drive West
Los Angeles, California 90045

CROSS-REFERENCE TO RELATED APPLICATIONS

This application claims the benefit under 35 U.S.C. §119(e) of the following co-pending and commonly assigned U.S. Provisional patent applications, which applications are incorporated by reference herein:

United States application serial number 60/159,069, entitled "MAPGUIDE PERSONAL DIGITAL ASSISTANT," filed on October 12, 1999, by Nemmara Chithambaram, et. al., attorney's docket number 30566.96USP1;

United States application serial number 60/193,141, entitled "SHARABLE SERVER UPLOADABLE REDLINING FOR PERSONAL DIGITAL ASSISTANT (PDA)," filed on March 29, 2000, by Nemmara Chithambaram, et. al., attorney's docket number 30566.97USP1;

United States application serial number 60/193,153 entitled "INDEXED RASTER VECTOR SCHEMA FOR PERSONAL DIGITAL ASSISTANT (PDA) DATABASES," filed on March 29, 2000, by Nemmara Chithambaram, et. al., attorney's docket number 30566.109USP1;

United States application serial number 60/193,142, entitled "INTERPROCESS API AND GRAPHICAL USER INTERFACE FOR PERSONAL DIGITAL ASSISTANT (PDA) DEVICES AND APPLICATIONS," filed on March 29, 2000, by Nemmara Chithambaram, et. al., attorney's docket number 30566.110USP1; and

United States application serial number 60/193,862, entitled "MAPGUIDE FOR MOBILE DEVICES SERVER," filed on March 30, 2000, by Nemmara Chithambaram, et. al., attorney's docket number 30566.112USP1.

This application is related to the following co-pending and commonly-assigned patent application, which application is incorporated by reference herein:

United States Patent Application Serial No. 09/411,506, entitled "VECTOR-BASED GEOGRAPHIC DATA", by Gregory A. Roy, et. al., Attorney Docket No. 30566.17USC1, filed on October 4, 1999, which is a continuation patent application of United States Patent No. 5,966,135 issued on October 12, 1999 (Application Serial No. 08/757,706 filed on October 30, 1996), by Gregory A. Roy et al., entitled "VECTOR-BASED GEOGRAPHIC DATA".

United States Patent Application Serial No. xx/xxx,xxx, entitled "METHOD AND APPARATUS FOR PROVIDING ACCESS TO MAPS ON A PERSONAL DIGITAL ASSISTANT (PDA)", by Nemmara Chithambaram et al., Attorney Docket No. 30566.96USU1, filed on the same date herewith;

United States Patent Application Serial No. xx/xxx,xxx, entitled "GENERALIZED, DIFFERENTIALLY ENCODED, INDEXED RASTER VECTOR DATA AND SCHEMA FOR MAPS ON A PERSONAL DIGITAL ASSISTANT", by Nemmara Chithambaram et al., Attorney Docket No. 30566.98USU1, filed on the same date herewith; and

United States Patent Application Serial No. xx/xxx,xxx, entitled "GEOGRAPHICAL DATA MARKUP ON A PERSONAL DIGITAL ASSISTANT (PDA)", by Nemmara Chithambaram et al., Attorney Docket No. 30566.97USU1, filed on the same date herewith.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates generally to electronic maps and geographic information, and in particular, to a method, apparatus, and article of manufacture for obtaining/creating a set of maps.

2. Description of the Related Art

Computer implemented geographic information systems (GIS) are known in the art. Such GIS provide for the retrieval and display of geographic information (e.g., maps). A GIS is a system of software, hardware, and data that delivers geographic data (street maps, property boundaries, power transmission lines, etc.) along with any associated attribute information. It can show you where a street is and also tell you the street name, when it was last paved, whether it is a one-way street, etc. Using a GIS, a user can perform complex queries (from a web browser to a server) to discover such things as how many people live near the street, what their income level is, and what the zoning laws are. A GIS can operate on a network/internet wherein the geographic information is stored on a server and transmitted to a client/user where the information (map picture and other data) is displayed on a web browser.

For the client to properly display the geographic information, a computer system with the appropriate processing capabilities, software, and memory is required. For example, a client may be required to utilize a computer with a web

browser such as INTERNET EXPLORER or NETSCAPE NAVIGATOR and have a minimum of 10 megabytes of available memory. Additionally, to display the geographic data such that a user does not have to wait an inordinate amount of time to retrieve and load the data, an appropriate internet connection (e.g., a 28.8 Kbps (kilo bits per second) modem) and a computer system with significant processing power (e.g., a minimum speed of 100 megahertz) may be required.

Field/utility technicians such as gas company employees, salespersons, plumbers, insurance adjusters, or any type of employment that requires travel to different locations, often utilize or require access to maps and geographic information. Further, such technicians often need to interact with a map to obtain relevant information. For example, a plumber/contractor may want to determine where the main gas line or water line on a street is located. However, while out in the field, the technicians often do not have a network connection, and carrying a laptop or desktop computer is cumbersome and impractical. Thus, it is desirable to have a small (handheld) portable computing device with the capabilities to display and interact with geographic information both online and offline.

Prior art handheld computing devices (also referred to as palm PCs or personal digital assistants (PDAs)), are often used to access and utilize personal information. Many handheld computing devices are available in today's marketplace. Typically, handheld computing devices are only slightly larger than the size of one's palm (hence, the name palm PC) and have a small display screen for viewing a plethora of items. Software can be installed on a PDA to provide enhanced functionality. For example, a personal productivity tool may be installed

to provide access to a calendar, contacts, email, Internet browsing, audio books, and audio recording capabilities. Card slots may also be available to provide additional memory or other functions (e.g., a modem). Additionally, some PDAs have infrared (IR) ports for communication.

The PDA environment, however, poses several challenges for geographic information systems in terms of memory, storage, processor speeds, wireless transmission rates, and display attributes. For example, PDAs commonly only maintain 96K or less of memory, 2 Mb (megabytes) or less storage, a 13 MHz processor speed, and a black and white or gray scale display mechanism. Further, field technicians using a PDA need to view and interact with maps displayed. The high amount of map data coupled with low bandwidth offered by wireless transmission services and the slow processors result in unacceptable download times.

Additional server-side support is needed to enable a GIS on a PDA. There is a need to provide such server-side support without modifying/substantially modifying existing server technology.

Thus, there is a need for a geographic information system that overcomes the above described deficiencies.

To help better understand embodiments of the invention, it is useful to describe the data utilized in prior art geographic information systems. Prior art geographic information systems display map pictures that are generated using raster data. Raster data represents a map picture with points in a grid. For example, on an X-Y axis, there may be a thousand points in the X direction and seven hundred

and sixty points in the Y direction. Each of these points represents a color. For example, some computer systems enable each point to represent one of eight colors. A map picture is then created by determining a color for each point in the grid.

Each map picture is static in that portions of the map picture cannot change independently of each other. That is, a portion of a map picture cannot be modified while viewing the map picture. Thus, while a map image may contain several layers of information, the information is merely a picture with no live data. When a portion of the map picture is to be modified, queried, or to perform any GIS analysis, the browser must request more data from the main server and the entire map picture is replaced. For example, each map picture, such as one representing the United States (US), may contain several layers of information, such as states, counties, and streets. When a user is viewing a map picture of the United States and wishes to view the counties in a particular state, the map picture is replaced with another map picture that contains the additional information. Typically, the additional information is stored at a server computer and the map picture is displayed at a client computer. When the map picture that is displayed is to be modified, the additional information is downloaded from the server computer. Because this additional information is in the form of raster data, it is typically time-consuming to download.

Additionally, some computer systems display schematics generated from vector-based data in computer aided design (CAD) files. Vector-based data uses descriptions of elements of the schematic to create the schematic. For example, if the schematic contains a line segment, the CAD file describes the line segment with

an endpoint and a length. Moreover, these computer systems enable users to view data in CAD files from the Internet and Intranets. For instance, when a user at a computer system wishes to view a schematic, the computer system downloads all of the data in the CAD file for that schematic from a data storage device via the Internet. The CAD file typically contains data corresponding to different levels of design of the schematic. In some instances, a user may wish to view only some of the data in the files, for example, the highest level of design of the schematic. In this case, although only a portion of data is required to display the schematic requested by the user, the computer system has already downloaded all of the data in the CAD file. Because it is time-consuming to download all of the data, it is inefficient to do so when only a portion of the data is required to satisfy a user's request. Further, it may not be possible to download all or even a portion of the data onto a PDA.

When vector or raster-based data are needed, the information is typically created and downloaded upon request. Consequently, each time a map is requested, delays from obtaining and transmitting the requested map result.

SUMMARY OF THE INVENTION

Field technicians such as plumbers, the department of water and power, contractors, the gas company, etc. often require access and use of maps and geographical information for those maps. One or more embodiments of the invention provide for the use of such mapsets on a personal digital assistant (PDA).

A server provides added functionality that enables the use of such geographic information on a PDA.

Mapsets are an ordered collection of maps that reside on distributed databases on the internet. One or more embodiments of the invention provide for a parallel processing architecture on the server side to generate and assemble multiple maps into a single database. At the time of assembling the map into the database, the order of the maps within the mapset is preserved. Accordingly, the application takes advantage of a host system with multiple CPUs and latencies inherent in the processing of a complex map are accounted.

One or more embodiments of the invention provide for the efficient network synchronization of mapset databases. The creation of the complex mapset databases is separated from the transmission of the mapset to a client PDA device. The server is configured to receive alerts that create/delete/update the database on the server side (specific to a user profile). The database on the server side is pre-constructed when such alerts are received. The pre-constructed database is in a form that can be easily sent over a linear data stream, easily reassembled from the stream into a PDA database, and in a location that is specific to the user profile and located by a web server.

As a result of the design architecture, the task of network synchronization may be accomplished by utilizing an hypertext transfer protocol (HTTP) 'GET' of an existing database (e.g., the pre-constructed database) and assembling the database on the PDA client from the database friendly format.

BRIEF DESCRIPTION OF THE DRAWINGS

Referring now to the drawings in which like reference numbers represent corresponding parts throughout:

FIG. 1 schematically illustrates a hardware and software environment for the architecture in accordance with one or more embodiments of the invention;

FIG. 2 illustrates the components of a servlet in accordance with one or more embodiments of the invention;

FIG. 3 is a flow chart illustrating how a servlet responds to requests from a client in accordance with one or more embodiments of the invention;

FIG. 4 illustrates the flow of information between server and client components in accordance with one or more embodiments of the invention; and

FIG. 5 illustrates the asynchronous creation of map data by a servlet in accordance with one or more embodiments of the invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

In the following description, reference is made to the accompanying drawings which form a part hereof, and which is shown, by way of illustration, several embodiments of the present invention. It is understood that other embodiments may be utilized and structural changes may be made without departing from the scope of the present invention.

Overview

One or more embodiments of the invention provide for the parallel processing and creation of a mapset on multiple central processing units (CPUs) for use in a geographical information system on a personal digital assistant (PDA). Further, embodiments of the invention provide for the pre-construction of the mapset prior to receiving a request from a client, such that the mapset may merely be transmitted when requested by an application on the client PDA.

General Architecture

Hardware Environment

The use, on a PDA, of a modified MAPGUIDE GIS currently available from the assignee of the present invention is provided. The existing MAPGUIDE GIS is more fully described in co-pending United States Patent Application Serial No. 09/411,506, entitled "VECTOR-BASED GEOGRAPHIC DATA", by Gregory Andrew Roy, et. al., Attorney Docket No. 30566.17USC1, filed on October 4, 1999, which is a continuation patent application of United States Patent

No. 5,966,135 issued on October 12, 1999 (Application Serial No. 08/757,706 filed on October 30, 1996), by Gregory A. Roy et al., entitled "VECTOR-BASED GEOGRAPHIC DATA", both of which are fully incorporated by reference herein.

FIG. 1 schematically illustrates a hardware and software environment for the architecture in accordance with one or more embodiments of the invention. A typical distributed computer system 100 uses a network/Internet 118 to connect technicians utilizing clients such as a thin client 102 (e.g. a PDA, WINCE, or PALM device) or a thick client 104 (e.g., a computer system running a browser) to server computers 106.

A thick client 104 as utilized in the existing MAPGUIDE GIS may comprise a computer with a web browser (enhanced with a plugin or viewer) that communicates with MapGuide server 120 through web server 110 that enables the ability to retrieve data (e.g., raster data, spatial data format (SDF) data 126, attribute data 128, etc.) through a database management system (DBMS) 114.

A thin client includes three classes of devices: handheld personal computers (HPC), palm-held personal computers (PPC or PDA), and smart phones. Using these devices, a thin client 102 may not provide the full processing and memory capabilities as a thick client 104. For example, as described above with respect to PDAs, thin clients 102 often have memory less than 100K, storage of less than 2-4 MB, processor speeds of 13 MHz, and limited display attributes. Consequently, additional server 106 side support (e.g., parallel processing of map data, the creation of mapsets, more generalized display data, simplified project files, de-cluttering

services, and possibly server management of user state) may be utilized. A typical combination of resources may include a network/Internet 118 comprising the Internet, LANs, WANs, SNA networks, or the like, clients 102 and 104 that are PDAs, personal computers or workstations, and servers 106 that are personal computers, workstations, minicomputers, or mainframes.

The network/Internet 118 connects client computers 102 and 104 executing the appropriate software applications to server computers 106 executing Web servers 110 and applications such as web applications 122, MapGuide servers 120, and servlets 108, and data files such as traditional map window files (MWF) 124, spatial data 126, and attribute data 128. Servlets 108 may also be located within or part of web server 110. The server 106 and its components may also be referred to as a back office system. Such a back office system maintains access to corporate databases, synchronization utilities, etc. The Web server 110 is typically a program such as IBM's HyperText Transport Protocol (HTTP) Server or Microsoft's Internet Information Server. The servlet 108 interfaces between a thin client 102 and a MapGuide server 120 such that any additional processing required by a thin client 102 may be performed by the servlet 108 transparently from MapGuide server 120. Accordingly, to properly perform, thin client 102 must communicate with servlet 108 through web server 110. Further, for thick client 104 to properly perform and execute a MapGuide application, thick client 104 must communicate with MapGuide Server 120 through web server 110.

The servers 106 may also execute a web application 122 that utilizes a Common Gateway Interface (CGI) 112 (or Netscape Application Programming

Interface (NSAPI), Internet Server Application Programming Interface (ISAPI), etc.), which interfaces between the Web server 110 and a database management system (DBMS) 114 that may be utilized to retrieve relevant geographical data (such as SDF data, raster data, Open DataBase Connectivity (ODBC) data, etc.) from database 116.

Web server 110 provides access to MapGuide server 120 that enables traditional access for thick client 104 through DBMS 114 to spatial data 126 and attribute data 128 for the relevant maps. To provide the additional processing capabilities required by thin client 102, servlet 108 interacts with web server 110, MapGuide server 120, and the MWF files 124.

Generally, components 108-116, and 120-128 all comprise logic and/or data that are embodied in or retrievable from a device, medium, signal, or carrier, e.g., a data storage device, a data communications device, a remote computer or device coupled to the computer via a network or via another data communications device, etc. Moreover, this logic and/or data, when read, executed, and/or interpreted, results in the steps necessary to implement and/or use the present invention being performed.

Thus, embodiments of the invention may be implemented as a method, apparatus, or article of manufacture using standard programming and/or engineering techniques to produce software, firmware, hardware, or any combination thereof. The term "article of manufacture" (or alternatively, "computer program product") as used herein is intended to encompass logic and/or data accessible from any computer-readable device, carrier, or media.

Those skilled in the art will recognize many modifications may be made to

this exemplary environment without departing from the scope of the present invention. For example, those skilled in the art will recognize that any combination of the above components, or any number of different components, including different logic, data, different peripherals, and different devices, may be used to implement the present invention, so long as similar functions are performed thereby.

Software Embodiments

In accordance with the hardware descriptions, thick clients 104 are complete computer systems with web browsers and full processing capabilities. However, the hardware limitations of a PDA device necessitate software limitations. Accordingly, to enable a geographic information system on a PDA, a thin client 102 is provided. To accommodate the thin client 102, additional support on server 106 may be utilized. For example, server 106 may provide parallel processing of map data, the asynchronous creation of mapsets (a set of maps related to a single map file), generalized display data, simplified project files, de-cluttering services, and possibly server management of the user state.

The architecture of the invention includes a data model that combines static raster layers (static raster data for multiple layers) with live vector objects to deliver good display and download performance, and also provides interactive selectable objects.

Vector based maps (also referred to as "map layer data" consisting of geographic information/data for one or more layers) are served by a servlet 108 and

are an encoded and spatially indexed vector representation of the geographic data. Such vector maps provide for a more "interactive" display with flexible zooming on the client 102, highlighting, etc. Alternatively, the Scalable Vector Graphics (SVG) representation as proposed by the WorldWideWeb Consortium (W3C) may be utilized. SVG allows for three types of graphic objects: vector graphic shapes (e.g., paths consisting of straight lines and curves), images, and text. Graphical objects can be grouped, styled, transformed, and composited into previously rendered objects.

A display background (that is raster based) for the geographic data and display layers is managed as a multi-level library of raster tiles. The raster layers are composed from multiple vector layers on servlet 108, resulting in better download and display performance. The raster map on the PDA allows panning (virtual roaming paradigm), and zooming across multiple levels. A smart-cache on thin client 102 allows the swapping of compact tiles from the database to memory, in a manner appropriate to the device.

Details of Servlet 108 Architecture

The servlet 108 is a bridge between a client 102 and the services traditionally supplied by server 106 and MapGuide server 120 to client 104. In addition, servlet 108 is tailored to support an offline mode of operation of client 102. Further, servlet 108 is intended for operation within the context of a larger workflow application, both on client 102 and server 106.

Servlet 108 accommodates additional processing needed by PDA thin client

102. To take advantage of existing web server 110, web application 122, MWFs 124, and MapGuide server 120 technology, servlet 108 may be implemented using the Java programming language. Accordingly, a single code stream is utilized to implement the servlet 108 on multiple platforms. Alternatively, any programming language that provides similar capabilities may be utilized. A minimum set of constraints, beyond those provided by web server 110 and MapGuide server 120, and the servlet 108 framework provide scalability. For example, each client 102 request may be self-contained in that it is responded to by a different instantiation of the servlet 108.

Servlet 108 may reside in a web server 110 and processes and creates mapsets. Additionally, servlet 108 responds (by interacting with MapGuide server 120) to requests for a map set and spatial data 126 and attribute data 128 (through DBMS 114) related to map objects, from multiple thin clients 102. Servlet 108 processes the requests, performs the visualization and encoding and sends the results back to the thin clients 102 (through web server 110).

FIG. 2 illustrates the subcomponents of servlet 108. The request broker 224 listens to and coordinates requests from multiple clients 102 (through web server 110), and conveys the requests to the query processor 230. The query processor 230 processes the query using services provided by the servlet resident viewer services component 232. Query processor 230 also uses the visualizer component 228 and encoder component 226 to perform cartographic decluttering and encoding of the data sent back to the client 102. Visualizer component 228 performs configurable decluttering of data in a manner that is appropriate to the display of

thin client 102. Encoder component 226 compacts and encodes the objects being sent to the thin client 102. Servlet resident viewer services components 232 represents the servlet 108 resident part of viewer services and communicates with mapguide server 120. The map guide server 120 may be web server 110 or a component of server 106.

FIG. 3 is a flow chart illustrating how servlet 108 responds to a request for a map from a client 102. At step 300, client 102 requests (through web server 110) request broker 224 on servlet 108 to download new layers (for dynamic data). At step 302, the request broker 224 (after coordinating requests from other clients 102 if necessary) conveys the request to the query processor 230. The query processor 230 conveys the query to the servlet 108 resident viewer services 232 and obtains the results at step 304. The results are decluttered (if specified) using the visualizer 228 at step 306 and encoded for compaction using the encoder 226 at step 308. The results (after visualization and encoding) are sent back to the client 102 at step 310. Obtaining the result in step 304 and steps 306-308 may be performed prior to receiving a request from client 102. Such performance provides for the asynchronous pre-construction of a mapset for use by individual clients 102 on a per-user basis.

FIG. 4 illustrates the flow of information between server 106 and client 102 components in accordance with one or more embodiments of the invention. On the back-end (i.e., server 106), a dispatching application 402 updates a work order database with work orders 406 and sends requests to servlet 108 to build a data catalog for the maps associated with the work orders 406, as well as a list of HTTP

links 408 that may be jumped to from the maps. The request may provide for the creation, deletion, or update of a work order. The files get updated on a per user basis for every work order 406 that is dispatched. In performing this activity, servlet 108 acts as a client 102 with respect to MapGuide servers 120.

When a client synchronization occurs (i.e., when client 102 requests map data), database 418, browser 422, and client application 420 go through synchronization operations. For client application 420, red-lining data 412 may be uploaded, and the map data 408 is downloaded. The browser 422 synchronization is driven by a list of links 414 maintained by servlet 108.

In a separate asynchronous operation, a redlining application 404 on server 106 massages and integrates any resultant new redlining data 412 uploaded from the PDA client 102 application 420 into the traditional spatial data resources 126.

Using offline operation on client 102, work orders 406 are fulfilled. This may result in a change in the status of the work orders 406 on PDA client 102 and the possible collection of red lining data 412 in the field on PDA client 102.

A web server application 122 may upload the list of web links 414, obtain the relevant web pages 416 identified by the links and transmit the web pages 416 to browser 422 (through web server 110) on PDA client 102 for viewing.

The map data 408 may also be referred to as a mapset. A mapset consists of: an initial map that is displayed when a mapset is selected; a set of maps based on the same MWF as the initial map that can be zoomed to sequentially from the initial map; and all maps that appear as links on the initial and zoomed maps. Further, each of the maps in a mapset contain data (such as object name, geometry, and the

value of the link if an object has one) for every selectable object on the given map. The multiple maps of a mapset are necessary to provide zooming from the original map and also linking to other maps from the original and from any of the 'zoomed' maps.

Servlet 108 asynchronously creates the mapset/map data 408 for client 102. Accordingly, the mapset/map data 408 may be constructed on a per-user basis by servlet 108 prior to being requested by client 102. The maps/map data are maintained in a mapset that reside on databases that are distributed across the internet. As a result of the distribution and multiple maps, the creation of a mapset may lead to latencies in gathering the individual maps comprising the mapset and potentially heavy processing demand. Accordingly, the mapsets may be created in a parallel manner instead of serial wherein multiple processors (CPUs) or machines are utilized in fulfilling a request.

To accommodate such parallel processing , all of the maps for a given mapset are identified early in the request processing cycle. In other words, the maps in a mapset are identified when a request for a mapset is received. Unique keys for each map enable the identification of each map in a mapset. Separate threads (e.g., Java threads) are instantiated to produce the PDA version of each map 418. Each thread may execute on a different computer/CPU. Each component (e.g., raster data, vector data, and meta data) of each map is tagged with a unique key to enable the identification of each component. A transient database of all of the components from all of the maps is assembled and resides on server 106. Servlet 108 utilizes the transient database and the unique keys to assemble the

mapset that is transmitted to client 102.

FIG. 5 illustrates the asynchronous creation of a mapset/map data 408 by servlet 108. The invention may be utilized by utility, plumbing, construction, or other similar type industries wherein field technicians 504 perform repairs, installations, etc. A dispatcher 502 establishes one or more work orders 406 each day for each field technician 504. Accordingly, dispatcher 502 controls the dispatching application 402 wherein a work order 406 may be saved in a back-end database 508. Once a work order 406 is entered by dispatcher 502, the dispatching application 402 transmits a request to create, modify, add, or delete a map/mapset that contains the map data 408 for the work order 406. The request is transmitted across the internal network (e.g., intranet 510) to the server 106 (e.g., a hyper text transfer protocol (HTTP) server or web server 110). The request is forwarded to servlet 108 that creates, deletes, or modifies the map data 408 based on the request. Accordingly, the mapset 408 is pre-constructed by servlet 108 on a per-user 508 basis prior to the mapset/data 408 being requested by the user 508.

While the servlet 108 creates, deletes, or modifies the mapset 408, field technician 504 utilizes a synchronization application 506 on client 102 to issue a HTTP 'GET' request. The GET method retrieves whatever information (in the form of an entity) is identified by the Request-URI (Uniform Resource Identifier). If the Request-URI refers to a data-producing process, it is the produced data that shall be returned as the entity in the response and not the source text of the process, unless that text happens to be the output of the process. The GET request is transmitted across internet 118 to server 106 (e.g., an HTTP server or web server

110). The map data/mapset 408 identified in the request (i.e., by the URI specified) is then retrieved into client 102. Thus, the single file stream containing the map data/mapset 408 is retrieved using the 'GET' HTTP request. Accordingly, the mapset 408 is asynchronously pre-constructed in parallel on a per user 504 basis so that it can be automatically transmitted upon request.

To allow client 102 to operate offline, it is necessary to include the proper amount of context relative to the initial display of a map in the client 102 database 418. Database 418 may be implemented as a single file on client 102. In such an embodiment, the first record may include a work order 406 table containing the number of work orders 406 in the database 418, and information regarding each work order 406 including the name of the work order 406, date and time stamp information for each work order 406, and a reference to the location of the map for the work order.

The second record of such a database 418 may contain a map entry table containing the number of work orders in database 418, a reference to the URI for each map entry and a reference to the first record for the map identified by the map entry. The first record for a map includes the total number of records used for the map, the number of tiles comprising the map, width and height information for the tiles, and information regarding the work orders that use the map. The second record of a map includes layer information (e.g., the number of layers and layer definitions that identify the type of object and display attributes). Following the second record of a map is a record for each raster tile and a record for each layer (comprising the number of objects in each layer, and information relating to each

object including the object bounds, the number of subobjects, a coordinate list of the vertices of the object, the object name, etc.). By formatting the data of a mapset 408 in this manner, the mapset 408 may be easily transmitted over a linear data stream, may be easily reassembled from the stream into a database 418 at client 102, and is in a location that is specific to the user profile and located by a web server 110.

The map database 418 for each user is pre-constructed in parallel on multiple servers 106 or CPUs within a server 106, ahead of its actual download by client 102. Thus, the construction of the mapset 408 is asynchronous relative to the synchronization request from the client 102. Such a design addresses the common occurrence wherein field technicians 504 tend to download their daily assignments at the beginning of the day. Further, the mapset may be modified as work orders 406 are added or deleted from a particular user 504. Accordingly, a list of work orders may be processed at a single time, instead of one at a time. Alternatively, the mapset 408 may be created synchronously with respect to the synchronization application 506.

Servlet 108 maintains a set of files containing a catalog database 418 downloaded by client 102, a timestamp of the last update to the catalog database 418, a list of non-map object links 414 referenced by the maps 408 in the catalog database 418, and a set of reference tables used to maintain the catalog database 418 on server 106.

One or more embodiments of the invention provide a shift of some of the work from client 102 to servlet 108, to transition from a client 102 request to

synchronize a bookmark to a client 102 request to synchronize a user 504, and to allow preparation of the request's response to take place ahead of the request.

As described, server 106 assumes responsibility for delivering an exact analog of a PDA catalog 418 containing the mapset 408. Servlet 108 also provides the ability to pre-construct (in parallel) a mapset 408 (to be used as a PDA catalog 418 on client 102) on a per-user basis by generating and assembling multiple maps into a single catalog 408 in parallel. The content of the data being synchronized changes from the mapset 408 for one or more bookmarks to the mapsets associated with all of a user's 504 current work orders 406.

In one or more alternative embodiments, client 102 may not store map data and constructs an image for display dynamically from vector data transmitted from server 106. In yet another embodiment, a map may consist of one raster image for a master or index map, along with multiple raster images to support a one-level zoom from that map. In such an embodiment, server 106 may construct and transmit a raster representation of a map upon client's 102 request and client 102 may store the data. Further, data on client 102 may be updated to cover all work orders 406 for a user 504. Alternatively, instead of updating the data on client 102, all of the data may be replaced.

Conclusion

This concludes the description of one or more embodiments of the invention. The following describes some alternative embodiments for accomplishing the present invention.

For example, any type of personal digital assistant or computer, such as a mainframe, minicomputer, or personal computer, or computer configuration, such as a timesharing mainframe, local area network, standalone personal computer, WINCE device, etc. could be used with the present invention.

In summary, a parallel processing architecture for complex mapset requests and the synchronization of map databases are provided in accordance with one or more embodiments of the invention.

The foregoing description of one or more embodiments of the invention has been presented for the purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise form disclosed. Many modifications and variations are possible in light of the above teaching. It is intended that the scope of the invention be limited not by this detailed description, but rather by the claims appended hereto.